Mariangela Orme

Executive Architect – IBM Rational

Rational Agile Day

Agile -- La sfida per un nuovo modo di sviluppare software

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano

Challenges to effective software delivery today

Complexity Challenges

- More granular service functionality in composite business applications
- Large number of projects and assets including custom, outsourced and packaged

Team Challenges

- Geographically dispersed teams that often include business partners
- Effective cross-organizational visibility and synchronization, sharing becomes an imperative

Process Challenges

- Need for market experimentation
- Blind adherence to process insensitive to potential business trade-offs
- Need for agility *at scale*

Tools Challenges

- Lack of standards impacts ability to collaborate, automate and report across teams and assumptions
- Frequent asset updates and changing interdependencies

How do I control this new world to gain advantage?

How Do these Challenges Affect Us?

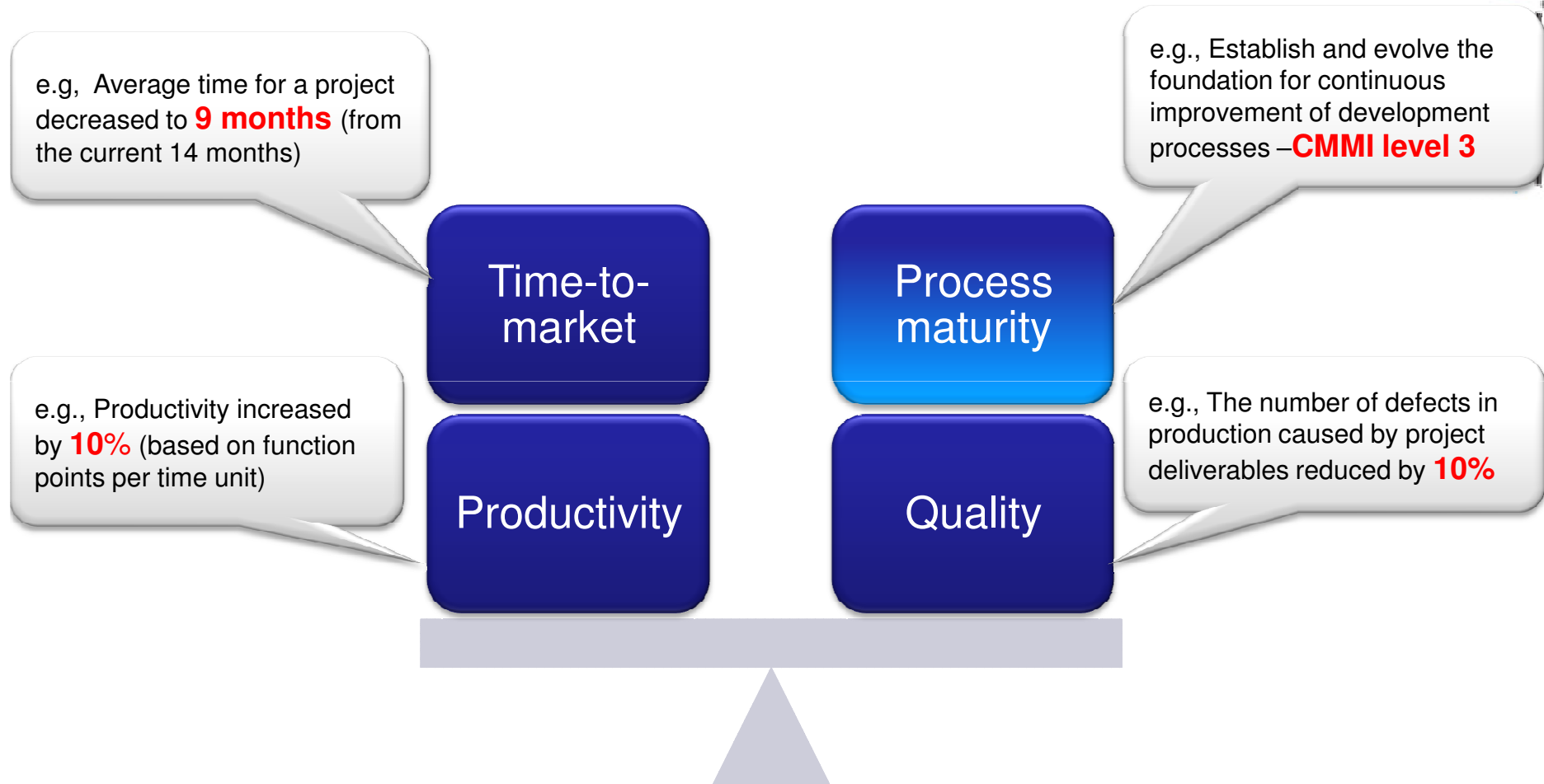From deliver all at one time → …to deliver less sooner

From technology driven solutions → …to business driven solutions

From software development → …to software delivery

From too little or too much process with blind adherence → …to right-sized process aligned with desired result

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano

We Need a Balanced Focus to Business Goals

e.g, Average time for a project decreased to **9 months** (from the current 14 months)

e.g., Establish and evolve the foundation for continuous improvement of development processes –**CMMI level 3**

Time-to-market

Process maturity

e.g., Productivity increased by **10%** (based on function points per time unit)

e.g., The number of defects in production caused by project deliverables reduced by **10%**

Productivity

Quality

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano
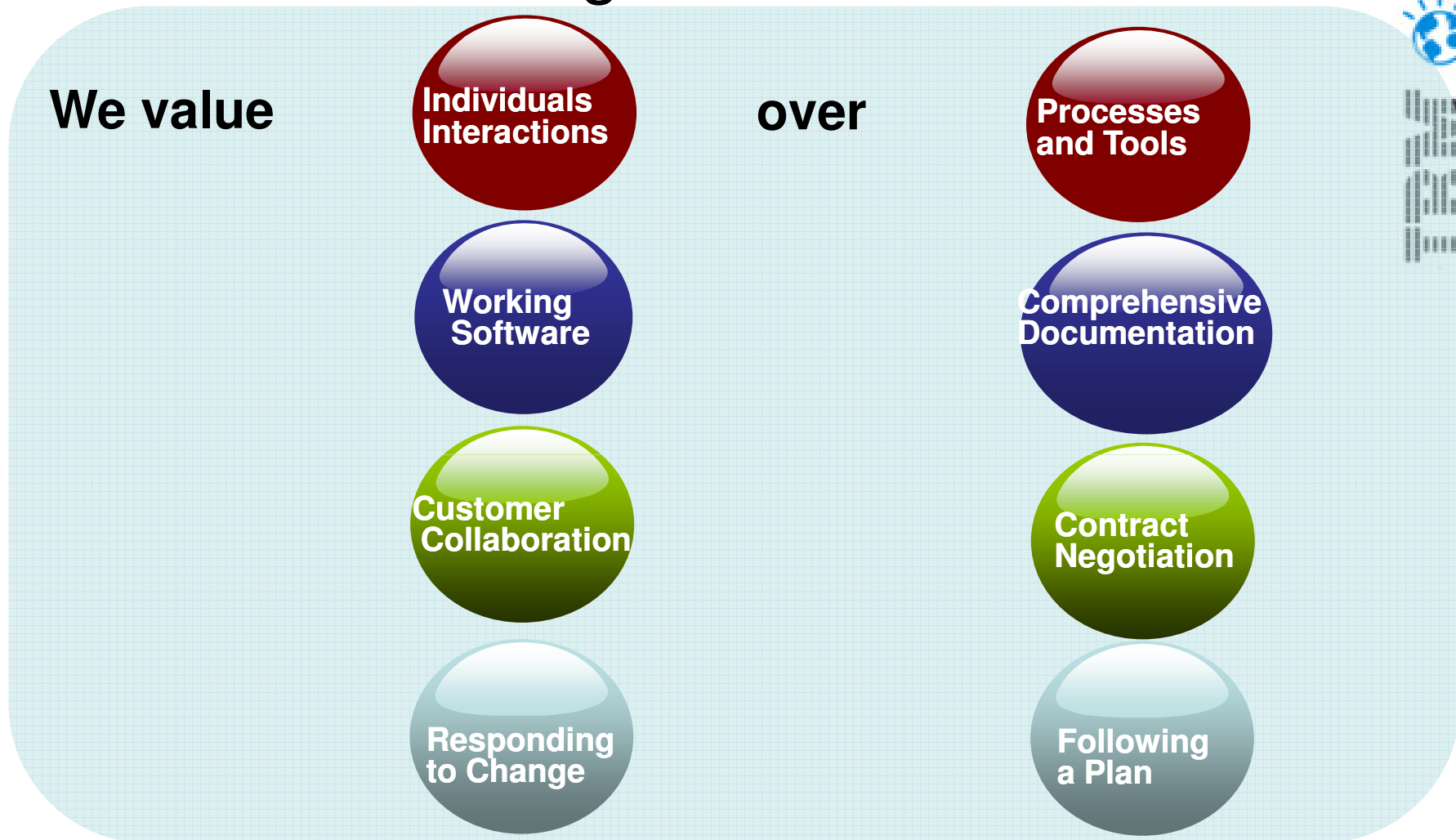
What is Agile?

- An iterative and incremental (evolutionary) approach performed in a highly collaborative and self-organizing manner with just the right amount of ceremony to frequently produce high quality software in a cost effective and timely manner which meets the changing needs of its stakeholders.

- Core principles
  - "Fits just right" process
  - Continuous testing and validation
  - Consistent team collaboration
  - Rapid response to change
  - Ongoing customer involvement
  - Frequent delivery of working software

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Agile values

We value   **over**

We value		over	
Individuals Interactions		**Processes and Tools**	
Working Software		**Comprehensive Documentation**	
Customer Collaboration		**Contract Negotiation**	
Responding to Change		**Following a Plan**	

While there is value in the items on the right, we value the items on the left more.

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Source: www.agilemanifesto.org

Rational Agile Day, 3 Marzo Milano

The Development Organization's View

Developer freedom

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

The Customer's View
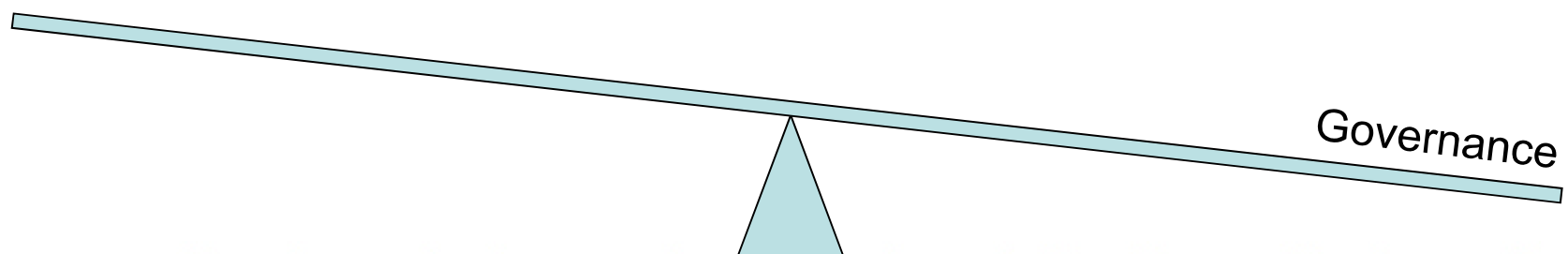
Governance

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano

I falsi Miti dell'Agile Software Development

- Non richiede pianificazione

- Non richiede documentazione

- Non richiede training

- Non è prevedibile

- Non è scalabile

- Non c'è disciplina

- Non occorre controllare il processo

Falso!

- *Il piano è flessibile e adattabile*

- *Si produce solo la documentazione che porta valore al progetto*

- *Occorre training e mentoring per gestire i team Agile*

- *La pianificazione è a più livelli. Solo il piano a breve termine è prevedibile in dettaglio. Il piano a lungo termine è più reattivo alle variazioni*

- *Per scalare, un team Agile richiede un maggiore livello di cerimonia di un team piccolo e co-localizzato*

- *Per collaborare in team e rispondere rapidamente ai cambiamenti occorre una certa disciplina*

- *Non c'è un processo "a taglia unica". Il processo deve essere adattato al contesto di ogni progetto*

How Agile is Different?

- Focus on collaboration:
  - Less paperwork and more conversation
  - Stakeholders actively involved

- Focus on quality:
  - We have a full regression test suite for our systems
  - We develop loosely-coupled, highly cohesive architectures
  - We refactor to keep them this way

- Focus on working software:
  - Greater feedback makes agile projects easier to manage
  - Less documentation is required
  - Less bureaucracy

- Agilists are generalizing specialists:
  - Less hand offs between people
  - Less people required
  - Specialists find it difficult at first to fit into the team

- Agile is based on practice, not theory:
  - This is a significant change from traditional development methods

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Agility is Relative

Organizational Drivers

Team Size
Geographical Distribution
Organization Distribution
Entrenched process, people, policy

- Mature or existing projects
- 50+ developers
- Complex, multi-platform applications

Agility at Scale

Dealing with Complexity

- Maturing p...
- Multi-platform
- Growing in complexity
- Remote or offshore work
- Greater need for coordination and handoffs

- Small team
- New projects
- Simple application
- Co-located
- Minimal need for documentation

Technical and Regulatory Drivers

Compliance
Governance
Application complexity

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

There is No Single Agile Process

Organizational Drivers

Team Size
Geographical Distribution
Organization Distribution
Entrenched process, people, policy

Rational Unified Process

Eclipse Way

XP, Scrum,
OpenUP, DSDM

Technical and Regulatory Drivers

Compliance
Governance
Application complexity
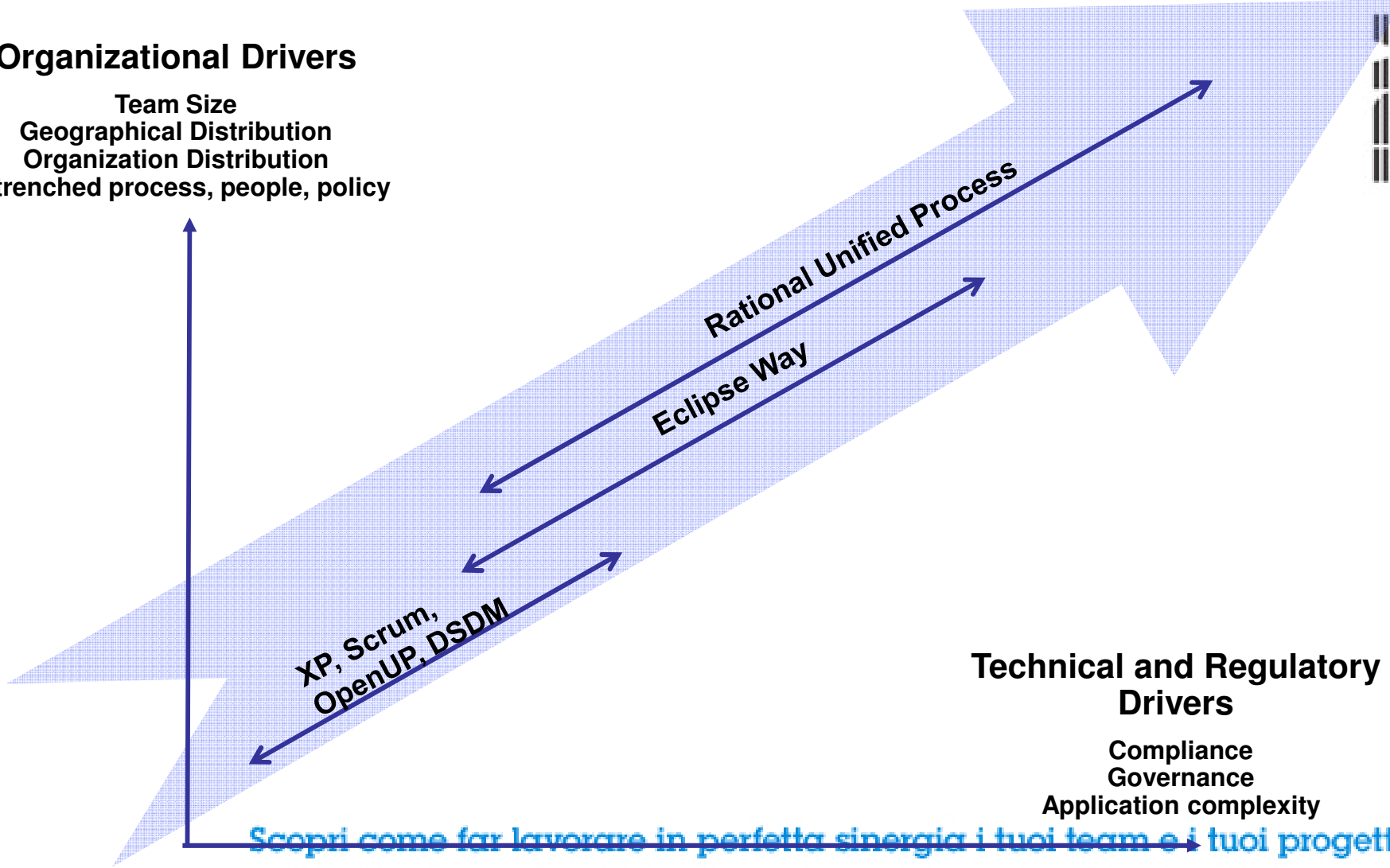
Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

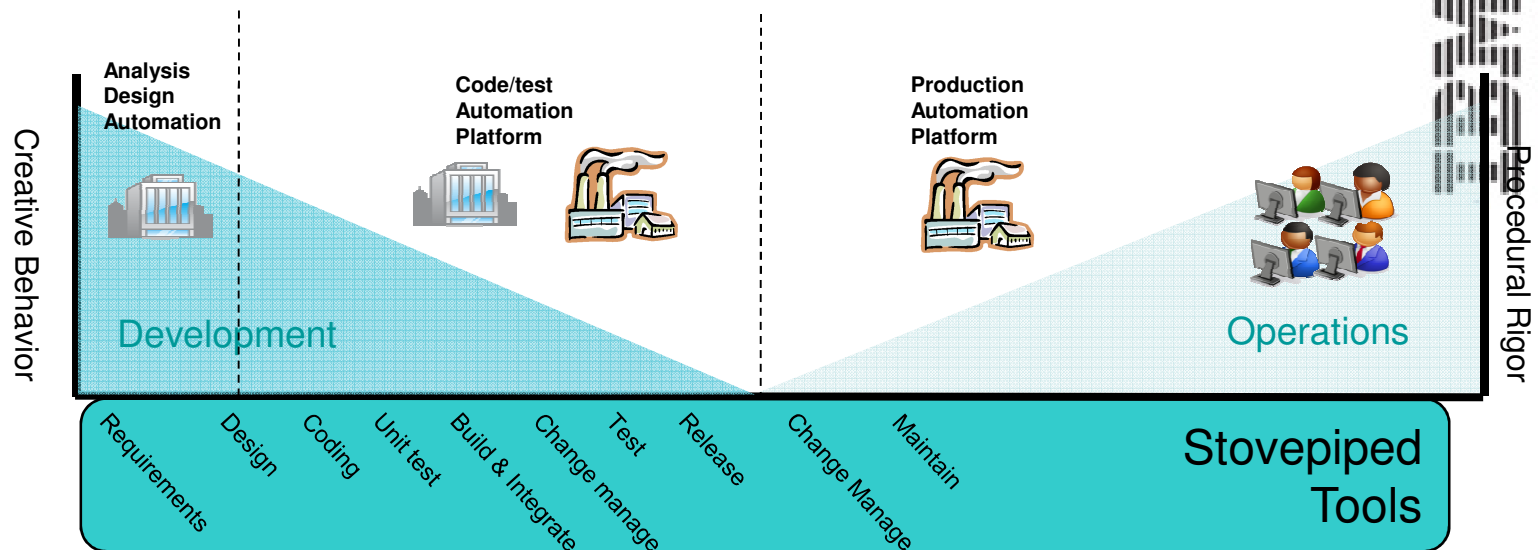# Adopting agile development practices help us achieve the defined goals
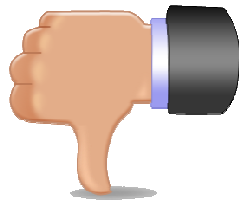
- Relative proportion of effort used on management is too high
- Requirements definition and the requirements sign-off process is too lengthy
- The change request process is cumbersome and time consuming
- Integration of the different solution elements is infrequent and done late in the project lifecycle
- A considerable "Hardening period" for the solution is needed just before and just after release in order to achieve an acceptable level of quality
- Further potential for reuse, consolidation and automation exists

- A high degree of efficiency in the development effort (improvement of 10% or more compared to today)
- Focus on the most revenue generating or cost saving deliverables
- Higher quality solutions provided at a cost that is the same or lower than previously
- Sourcing options (including reuse potential) identified and leveraged
- Deliver solutions to the business within short releases (always <12 months but typically releases every 3-4 months for agile projects)
- Risk associated with project business deliverables and project schedule are identified and mitigated early and efficiently
- Ability to change scope as required with minimal impact on schedules and commitments
- Ability to change schedules as required with limited notice

Rethinking Software Delivery

Waterfall Process Platform

Creative Behavior

Procedural Rigor

Analysis Design Automation

Code/test Automation Platform

Production Automation Platform

Development

Operations

Requirements | Design | Coding | Unit test | Build & Integrate | Change manage | Test | Release | Change Manage | Maintain

Stovepiped Tools

Agile Delivery Process Platform

Creative Behavior

Procedural Rigor

Analysis Design Code Automation

Software Delivery Automation

Consumers Users

Development

Common Software Delivery Automation

Operations

Implications

Automation
Measurement
Close Customer Relationship
Project visibility

Collaborative Automation Platform

Sco... tuoi team e i tuoi progetti.
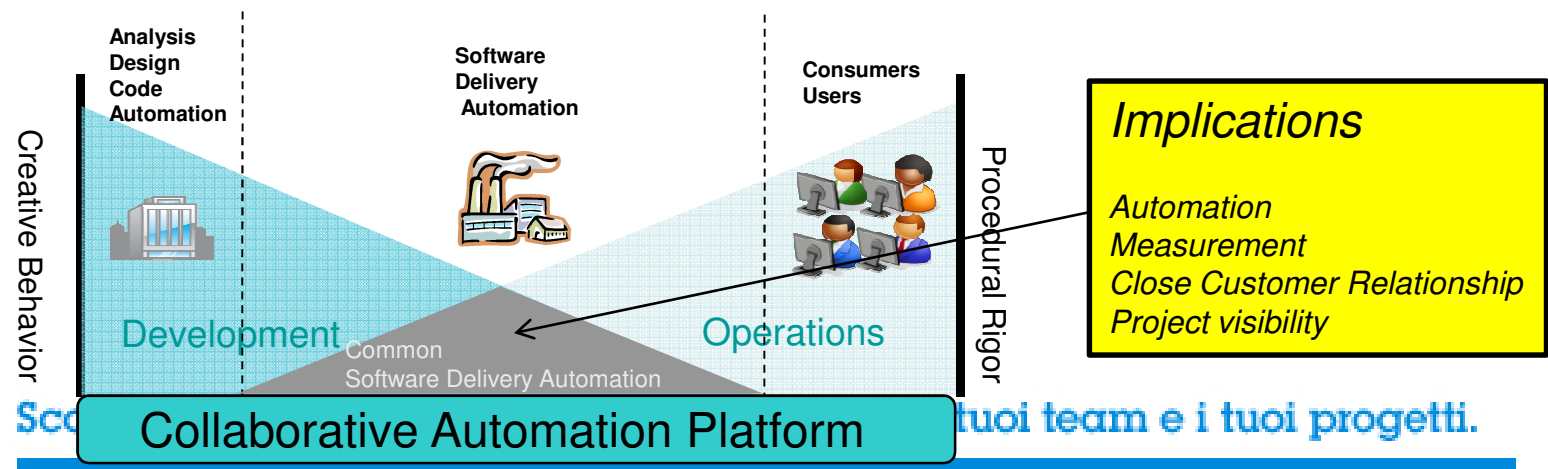
Application life-cycle management

The coordination of development life-cycle activities, including <u>requirements, modeling, development, build</u>, and <u>testing</u>, through:

1) ***Process Automation*** *- Enforcement of processes that span these activities.*

2) ***Traceability*** *- Management of relationships between development artifacts used or produced by these activities.*

3) ***Reporting*** *- Reporting on progress of the development effort as a whole.*

• ALM doesn't support specific life-cycle activities; rather, it keeps them in sync.

• ALM doesn't live in practitioner tools; it lives in the integrations between them.

• A collection of development life-cycle tools is not necessarily ALM. They need to be integrated

The Business Promise of ALM is Widely Anticipated

- Companies acquire ALM tools with the best of business-centric aspirations
  - Higher quality
  - More customer satisfaction
  - Aligning business and IT
  - Faster time to market
- Motives internal to IT also feature
  - Lower costs/higher productivity
  - More predictable delivery
- So every vendor wants to stake a claim in the ALM value proposition

What value was your organization looking to capture by purchasing and implementing these tools? (select all that apply)

Category	Percentage
Better quality	96%
Better customer satisfaction	84%
Better compliance with your company's standards (eg. documentation requirements)	84%
Better business-IT alignment	82%
Lower costs	82%
Better predictability	80%
Faster time-to-market	74%
Better compliance with third parties' rules and regulations (eg. SOX)	60%
Other	4%

Base: All respondents

Source: "ALM/PPM Tool Utilization, A commissioned study conducted by Forrester Consulting on behalf of Wipro Consulting Services, July 2008

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

For the full study cited on this slide, see http://wipro.us/pdf/whitepaper/alm_ppm_tools_investments_gone_awry.pdf

Rational Agile Day, 3 Marzo Milano

The Business Value Outcome is Often Disappointing

- Only 38% of surveyed respondents believe they **captured the expected value** from ALM tools.
- The other 62% were disappointed
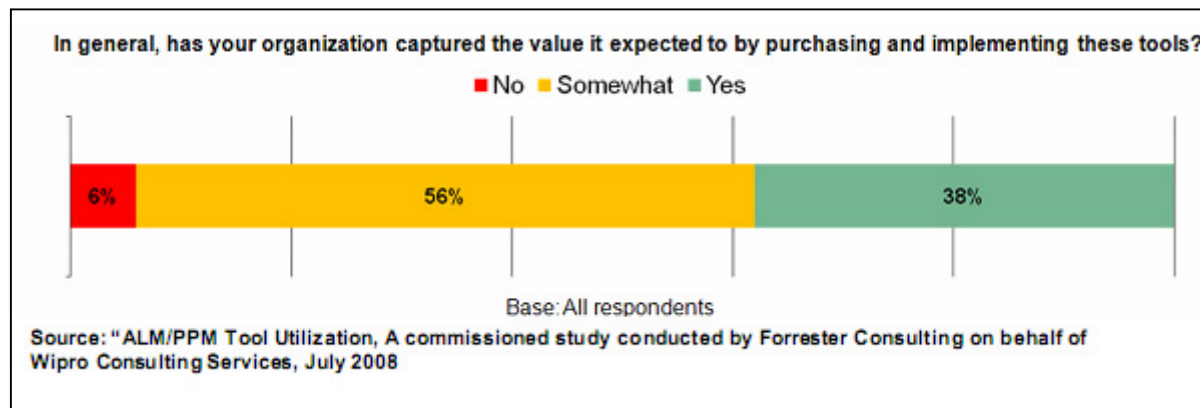
In general, has your organization captured the value it expected to by purchasing and implementing these tools?

■ No ■ Somewhat ■ Yes

| 6% | 56% | 38% |

Base: All respondents

Source: "ALM/PPM Tool Utilization, A commissioned study conducted by Forrester Consulting on behalf of Wipro Consulting Services, July 2008

For the full study cited on this slide, see http://wipro.us/pdf/whitepaper/alm_ppm_tools_investments_gone_awry.pdf

There are Multiple Reasons for the Disappointment

What prevents your organization from getting more value out of these tools? (select all that apply)

Resources
- We are distracted by day-to-day delivery pressures. — 78%
- We lack the necessary manpower. — 48%

Learning
- We lack the necessary internal expertise. — 56%
- The tools are too full-featured, too complex. — 44%
- We provide insufficient training and support for these tools. — 40%
- We haven't received enough training and support on these tools. — 30%

Process
- We have many conflicting processes. — 48%
- We don't have processes in place. — 26%
- The tools don't fit our processes. — 26%

Technology
- The tools don't integrate properly with other tools. — 62%
- The tools are too hard to customize. — 42%
- The tools are too buggy. — 18%
- The tools are too lightweight, don't fit our needs. — 12%

Culture
- We don't believe in the value of these tools. — 12%
- We don't believe it is possible to get more value from these tools. — 10%

Base: All respondents

These Problems are at least 20 Years old
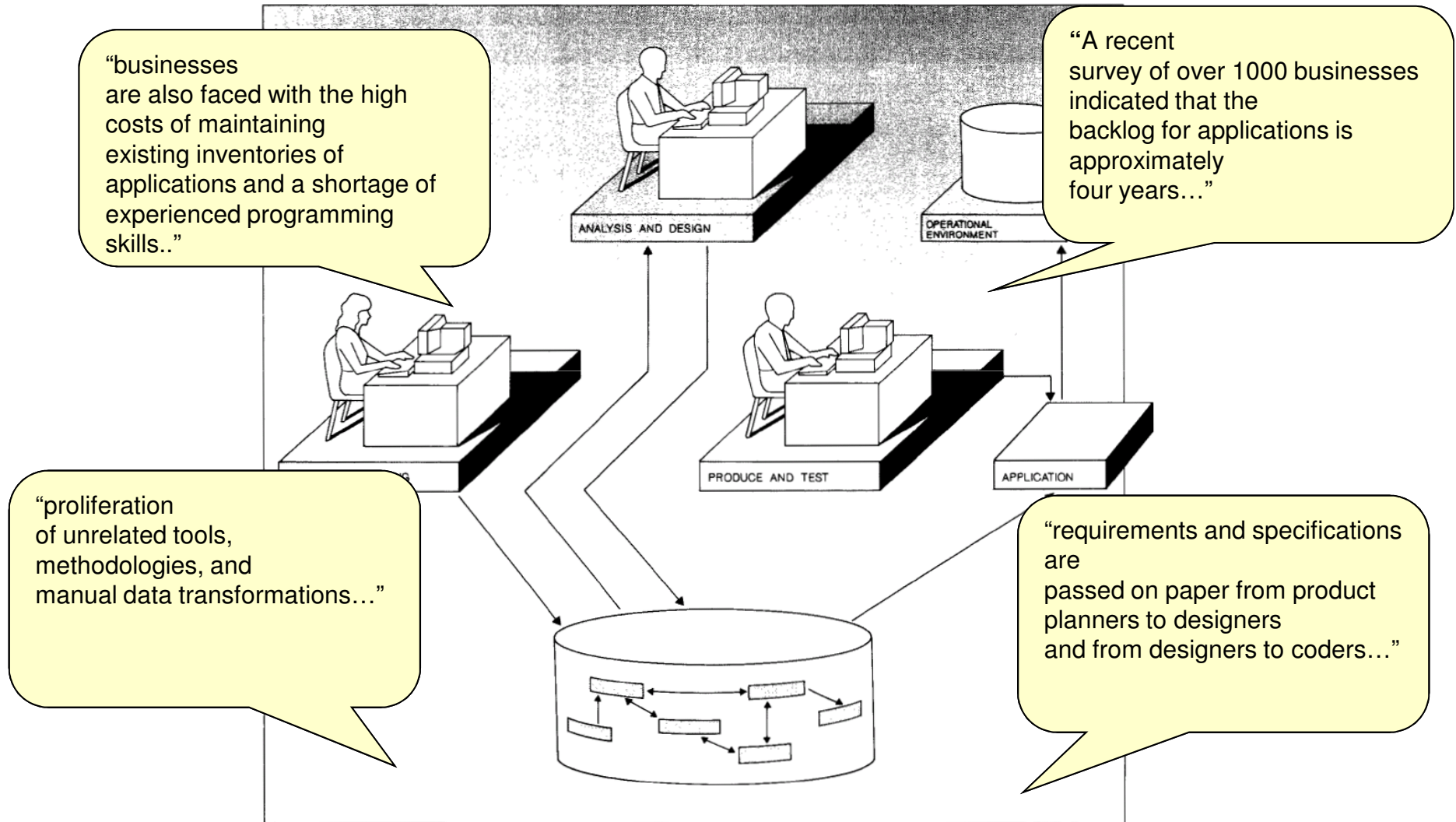
Source: Presentation on IBM's AD/Cycle, circa **1990!**

Traditional Approaches To ALM Integration Have Fallen Short

Limited choice and coverage

Single repository
- Hard to add existing (legacy) tools
- Difficult to evolve tools individually
- Limited to a single vendor's tools or affiliates

Monolithic Repository
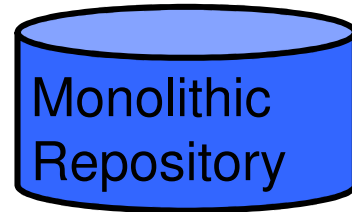
Point-to-point integrations
- Limited coverage: there are too many tools to cover more than a small fraction of possibilities
- Tight dependencies between tools require lockstep upgrades
- Proprietary APIs create vendor lock-in
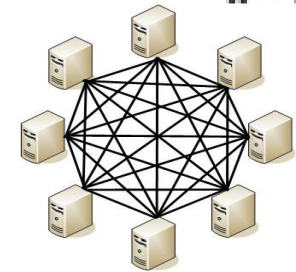
Universal metadata standard
- Too slow to complete to keep pace with the market
- Hostage to vendor in-fighting
- Difficult to migrate existing project data and assets
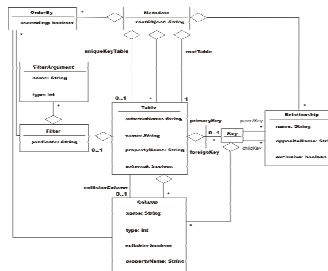
Standard implementations
- Requires "forklift" rip and replace of existing tools
- Hard to get widespread vendor support
- Insufficiently flexible to address different user approaches
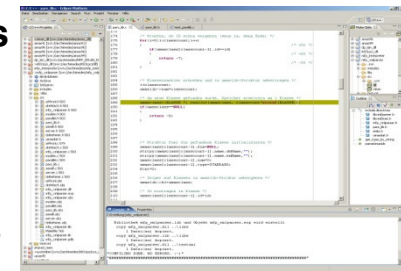
Slow to emerge and disruptive to adopt

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano

Jazz is Something Different
An Open and Extensible Software Development Platform
Supporting _Collaborative_ ALM
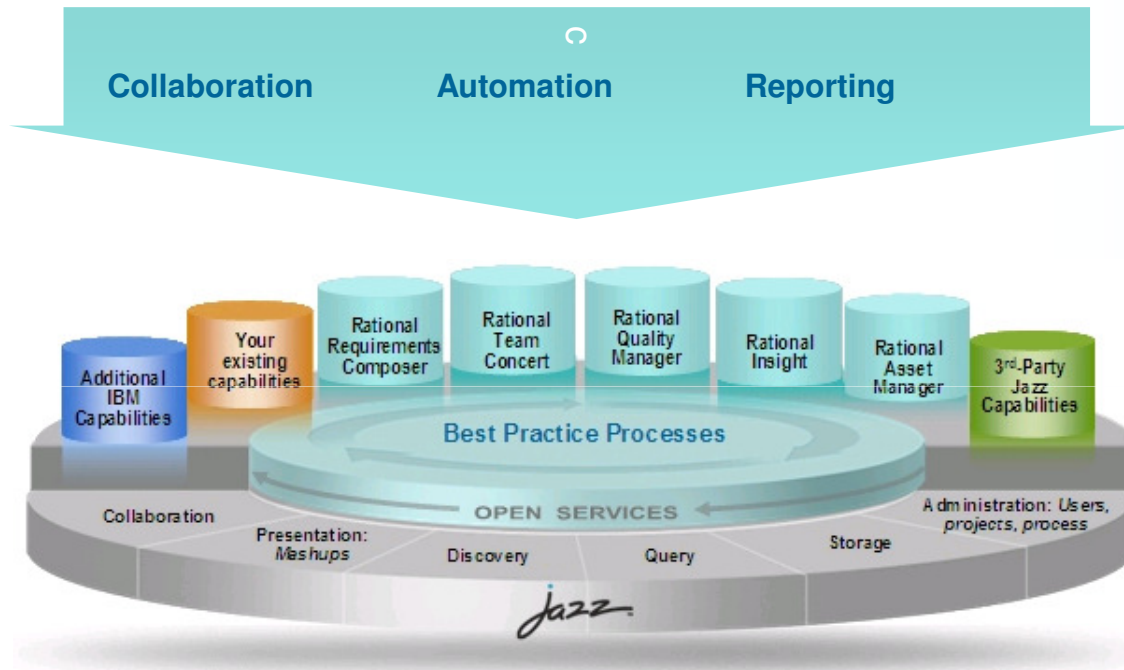
Provides

- A scalable, extensible team _collaboration_ platform

- End-to-end, artifact traceability

- Flexible and configurable team-specific process

- Integrated collaboration around the lifecycle artifacts

- Access to real time information for decision making

Jazz is a project & software delivery platform for _transforming how people work together_ to deliver greater value & performance from software investments.

The Jazz Foundation Architecture

3rd Party Products integrating with Existing Jazz applications

3rd Party Products built using Jazz Foundation

3rd Party

3rd Party Product

REST API

Rational Quality Manager ↔ Rational Team Concert ↔ Requirements Composer

Jazz Products

REST APIs

Discovery	Workflow	Administration *(users, projects)*
Presentation	Process	Query
Collaboration	Additional Services	Storage

Jazz Foundation Services

(Additional Services may be added to the foundation)

22

First wave of products built on Jazz technology

Rational.
Requirements Composer
Business Expert Collaboration

Elicit, capture, elaborate, discuss and review requirements

Rational.
Rational Insight
Cross-project and -team reporting
Performance management and measurement for integrated lifecycle intelligence

Rational.
Team Concert
Collaborative software delivery

Collaborative SCM, work item, build automation & iteration planning

Rational.
Quality Manager and Test Lab Manager
Lifecycle quality management
Coordinate quality assurance plans, processes and resources

Future IBM Capabilities

Your existing capabilities

Business Planning & Alignment

Product & Project Management

Collaborative Lifecycle Management

Compliance & Security

Engineering & Software Tools

3rd-Party Jazz Capabilities

Best Practice Processes

OPEN SERVICES

Collaboration

Presentation: *Mashups*

Discovery

Query

Storage

Administration: *Users, projects, process*

jazz™

IBM Business Partner

VMLogix

Mainsoft

VIRTUAL LABS SURGIENT

QSM

cm:logic

iRise

sourceIQ

RAVENFLOW

CAST

blackduck

WebLayers

SUBVERSION

23

Jazz Enables Collaborative ALM

- Win-Win Tool Adoption
  - Development more effective
    - Enables Team to Focus on their Day Job, rather than Producing Status Presentations (fosters grass roots adoption)
  - Organization more effective
    - Access to real-time project status enables reality-based decisions (encourages top-down support)

- Grass Roots adoption within IBM
  - Over 60 Rational development teams using Rational Team Concert
  - More than 125 other teams around IBM
    - WebSphere Development, Lotus Development, Tivoli Development, Research, GBS, etc

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Rational's Collaborative ALM
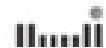
- "I'm Bob the **Product Owner**."

I am responsible for producing, prioritizing and accepting user stories.

- "I'm Scott the **Scrum Master**."

I coach the team, manage change, and track health and quality by collaborating with my team

Marco, Deb **Dev Lead**. Developer

We're developers who embrace agile practices and continuous integrations.

- "I'm Tanuj, a **Tester** in the team."

I practice concurrent testing

Rational Requirements Composer

- Elicit stories
- Manage Collection
- Elaborate stories

Rational Team Concert

- Prioritize Backlog
- Team Plans Iteration
- Detail sprint plans
- Source/fix developed, tested, delivered
- Triage defects / change requests
- Sprint Review

Rational Quality Manager

- Plan & Align Test Effort
- Detail test plans
- Test. Submit/ Confirm

2 Plan   **3 Sprint**   **4 Deliver**

Rational Agile Day, 3 Marzo Milano

Delivering real-time, global access to project information

Example: requirements

Developer
Change / Config Mgmt
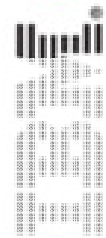
Analyst

Requirements

Tester
Quality Management

Developers link to requirements from work-items
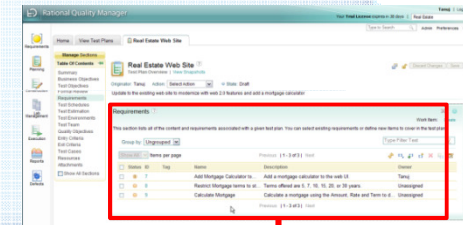
Testers link to requirements from test plans and test cases

Analysts communicate requirements with links to development and test plans

Scopri come ~~f~~ a i tuoi team e i tuoi progetti.

Rational Agile Day, 3 Marzo Milano
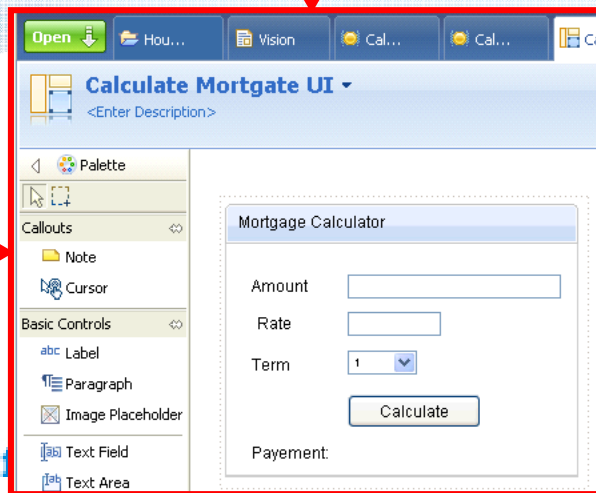
Accelerating in-context workflow
Example: Log a defect in context of test failure and requirements
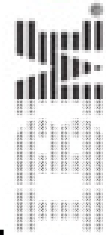
Analyst
Requirements

Developer
Change / Config Mgmt

Tester
Quality Management

Test Execution Results link to defects
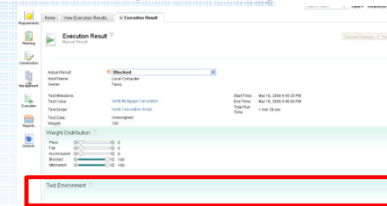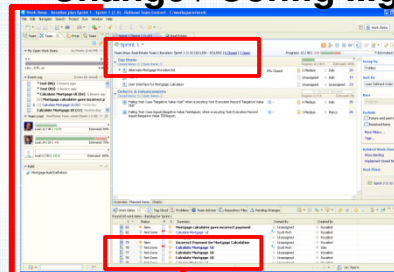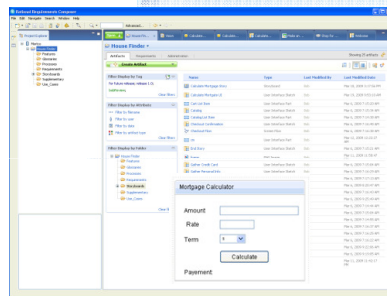
Defects can link to requirements
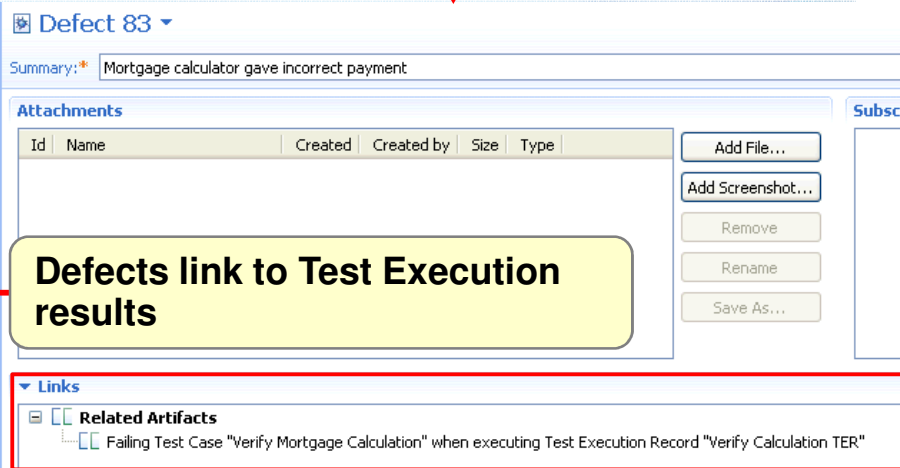
Defects link to Test Execution results

Collaborate using Workitems and Plans

Check the project status and health

www.jazz.net - Transparent development visibility

Suppose we did our development out on the Internet?

- A transparent software delivery laboratory where you can...

  - Communicate with the development team

  - Track the progress of builds and milestones

  - Get the latest product trials and betas

  - Join developers and product managers in discussion groups

  - Submit defect and enhancement requests

Scopri come far lavorare in perfetta sinergia i tuoi team e i tuoi progetti.

Questions